IN THE CLAIMS

Please amend the claims to the following.

1. (Currently Amended) An apparatus comprising:

a trigger-response mechanism that includes at least one bank of user-programmable registers to identify a user-defined trigger event; and

~~a~~ thread switch handler logic coupled to the trigger-response mechanism~~, the thread switch handler~~ to perform a light-weight thread switch from a first thread to a second thread ~~invoke a second instruction stream~~ responsive to ~~an indication from the trigger-response mechanism that~~ the user-defined trigger event ~~has~~ occuring[[ed]] during execution of the first thread ~~a first instruction stream~~.

2. (Currently Amended) The apparatus of claim 1, wherein the user-defined trigger event includes a synchronous user-defined trigger event~~, and wherein the thread switch handler is further to invoke the second instruction stream responsive to an indication from the trigger-response mechanism that the synchronous user-defined trigger event has occurred during execution of the first instruction stream~~.

3. (Currently Amended) The apparatus of claim 1, wherein the user-defined trigger event includes an asynchronous user-defined trigger event~~, and wherein the thread switch handler is further to invoke the second instruction stream responsive to an indication from the trigger-response mechanism that the asynchronous user-defined trigger event has occurred during execution of the first instruction stream~~.

4. (Currently Amended) The apparatus of claim 1, wherein the thread switch handler logic to perform a light-weight thread switch from the first thread to the second thread comprises: ~~is to~~ saving[[e]] a[[n]] first instruction pointer address for the first thread ~~instruction stream~~ before setting a second instruction pointer address for ~~invoking~~ the second thread~~instruction stream~~.

5. (Currently Amended) The apparatus of claim 4, further comprising: a task queue to receive the first instruction pointer address.

6. (Original) The apparatus of claim 5, wherein: the task queue further comprises a memory location.

7. (Original) The apparatus of claim 5, wherein: the task queue further comprises a register.

8. (Original) The apparatus of claim 1, further comprising: a plurality of event counters coupled to the trigger-response mechanism, wherein each event counter is to detect an atomic processor event.

9. (Currently Amended) The apparatus of claim 8, wherein the user-defined trigger event includes an asynchronous trigger event based on one or more of the atomic processor events~~, and wherein the thread switch handler is further to invoke the second instruction stream responsive to an indication from the trigger-response mechanism that the asynchronous user-defined trigger event has occurred during execution of the first instruction stream~~.

10. (Currently Amended) The apparatus of claim 1, wherein the thread switch handler logic is to perform the light-weight thread switch from the first thread to the second thread transparently to an Operating System (OS) and without OS intervention. ~~the thread switch handler is to save context information for the first instruction stream before invoking the second instruction stream~~.

11. (Currently Amended) The apparatus of claim 10, wherein: the thread switch handler is further to save a light weight context for the first thread ~~instruction stream~~ in a memory location before performing the light-weight context switch from the first thread to the second thread ~~invoking the second instruction stream~~.

12. (Currently Amended) The apparatus of claim 10, wherein: the thread switch handler is further to save a light weight context for the first thread ~~instruction stream~~ in a register before performing the light-weight context switch from the first thread to the second thread ~~invoking the second instruction stream~~.

13. (Currently Amended) The apparatus of claim 1, further comprising:

one or more user-programmable control registers coupled to the thread switch handler logic;

the value of the one or more control registers to indicate a weight of the light-weight ~~context information for the~~ thread switch from the first thread to the second thread. ~~first instruction stream to be saved responsive to invoking the second instruction stream~~.

14. (Currently Amended) A system comprising:

a memory to hold an instruction; and

a processor coupled to the memory, the processor including raw event detection logic to detect at least one raw event, a user-addressable register to specify a user-defined trigger event based on the at least one raw event, and a switch handler to invoke a helper thread responsive to the occurrence of the user-defined trigger event.

15. (Previously Amended) The system of claim 14, wherein: the instruction includes a marking instruction, when executed, to specify the user-defined trigger event in the user-addressable register.

16. (Previously Amended) The system of claim 14, wherein: the instruction is a trigger instruction; and raw event detection logic is to detect an opcode of the trigger instruction when the trigger instruction reaches an execution phase of an execution pipeline.

17. (Previously Amended) The system of claim 14, wherein: the processor further includes a user-addressable control register to specify a weight of a context to be saved responsive to invoking the helper thread.

18. (Currently Amended) The system of claim 14, wherein: the switch handler is further to maintain minimal context information for a current thread before invoking the helper thread, wherein the minimal context information includes a context weight less than a full context weight by at least a weight of excluding[[es]] traditional context information.

19. (Original) The system of claim 18, wherein: the excluded traditional context information further comprises general register values.

20. (Original) The system of claim 18, wherein the minimal thread context information comprises an instruction pointer address value.

21. (Currently Amended) A method comprising:

in response to detecting a user-specified trigger condition;

suspending execution of a first thread on a [[single-]]threaded processor; and

utilizing hardware to switch ~~save minimal~~ context information of the first thread with a second thread ~~for the current thread~~ without operating system intervention, wherein the context information has a first weight that is user-defined in a user-addressable control register.~~; and~~

~~invoking a second thread on the single-threaded processor without operating system intervention.~~

22. (Original) The method of claim 21, wherein:

detecting a user-specified trigger condition further comprises determining that a trigger instruction has been encountered.

23. (Original) The method of claim 21, wherein:

detecting a user-specified trigger condition further comprises determining that an asynchronous condition specified in a marking instruction has been encountered.

24. (Currently Amended) The method of claim 21, wherein: the first weight includes only an instruction pointer address. ~~utilizing hardware to save minimal context information further comprises saving an instruction pointer address value.~~

25. (Previously Amended) The method of claim 21, further comprising:

determining that the first thread should be resumed;

restoring the minimal context information for the first thread; and

resuming execution of the first thread without operating system intervention.

26. (Original) The method of claim 21, wherein detecting a user-specified trigger condition further comprises:

receiving a marker instruction that specifies the trigger condition; and

monitoring a plurality of atomic event indicators to detect the trigger condition.

27. (Original) The method of claim 21, wherein detecting a user-specified trigger condition further comprises: generating an asynchronous response to indicate that the second thread should be invoked.

28. (Previously Added) A processor comprising:

event detection logic to detect a raw event;

user-programmable event logic coupled to the event detection logic to indicate a user-defined trigger event, the user-defined trigger event to be based on at least the raw event;

user-programmable context control logic to specify a weight of a context to be saved; and

thread switch logic coupled to the user-programmable event logic and context control logic, the thread switch logic, in response to the user-defined trigger event being detected, to save a portion of a first context based on the weight of a context to be saved that is to be specified in the user-programmable context control logic and to spawn a helper thread without operating system intervention.

29. (Currently Amended) The processor ~~system~~ of claim 28, wherein the user-programmable event logic includes at least a user-programmable event register, and wherein the user-defined trigger event is to be programmed in the user-programmable event register in response to execution of a user marking instruction.

30. (Currently Amended) The processor ~~system~~ of claim 28, further comprising trigger response logic coupled to the user-programmable event logic and the event detection logic to detect the user-defined trigger event based on at least the raw event, wherein the trigger response logic is to monitor for the user-defined trigger event for a predetermined timeout period after execution of the user-marking instruction.